

(19)日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11)特許出願公開番号

特開平8-96021

(43)公開日 平成8年(1996)4月12日

(51)Int.Cl.⁶

識別記号

庁内整理番号

F I

技術表示箇所

G 0 6 F 17/50
11/25

9191-5H

G 0 6 F 15/ 60
11/ 26

6 6 4 K
3 1 0

審査請求 未請求 請求項の数5 O L (全 7 頁)

(21)出願番号 特願平6-232785

(22)出願日 平成6年(1994)9月28日

(71)出願人 000005108

株式会社日立製作所

東京都千代田区神田駿河台四丁目6番地

(72)発明者 鈴木 薫

神奈川県秦野市堀山下1番地 株式会社日立製作所汎用コンピュータ事業部内

(72)発明者 中山 和重

神奈川県秦野市堀山下1番地 株式会社日立製作所汎用コンピュータ事業部内

(72)発明者 米山 修二

東京都江東区福住1丁目13番12号 株式会社日立東サービスエンジニアリング内

(74)代理人 弁理士 小川 勝男

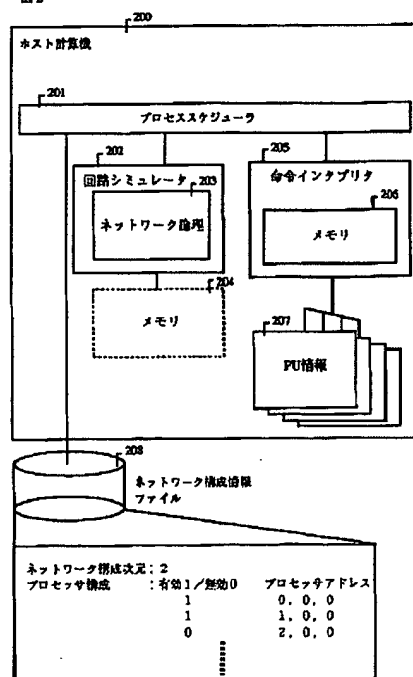
(54)【発明の名称】 並列プロセッサ論理シミュレーション装置

(57)【要約】 (修正有)

【目的】 命令インタプリタでプロセッサ制御論理部をシミュレートし、ネットワーク網論理のみを回路シミュレータで行い、回路シミュレータの論理実装制限に関係無く、大規模な並列プロセッサシステムの論理シミュレーションを効率良く高速に実施。

【構成】 ネットワーク網で接続される並列プロセッサの論理シミュレーション装置において、ネットワーク網論理とネットワーク制御論理部をゲートレベルでシミュレーションする回路シミュレータ202と、プロセッサ制御論理部をデータ転送機能レベルでシミュレーションする命令インタプリタ205と、回路シミュレータ202と命令インタプリタ205をイベント発生により制御するプロセススケジューラ201と、ネットワーク構成情報が登録されプロセススケジューラ201からアクセスされる第1の記憶部と、各プロセッサユニット情報が登録され命令インタプリタ205が管理する第2の記憶部とよくなる。

図2



1

【特許請求の範囲】

【請求項1】複数のプロセッサユニットがネットワーク網で接続される並列プロセッサの論理シミュレーションを行なう論理シミュレーション装置であって、前記ネットワーク網論理及び前記プロセッサユニットのネットワーク制御論理部をゲートレベルでシミュレーションする回路シミュレータと、前記プロセッサユニットのプロセッサ制御論理部をデータ転送機能レベルでシミュレーションする命令インタプリタと、前記回路シミュレータ及び前記命令インタプリタをイベント発生により制御するプロセススケジューラと、前記ネットワーク網のネットワーク構成情報が登録され前記プロセススケジューラからアクセスされる第1の記憶部と、前記複数のプロセッサユニットの各プロセッサユニット情報が登録され前記命令インタプリタが管理する第2の記憶部とから成ることを特徴とする並列プロセッサ論理シミュレーション装置。

【請求項2】1つの命令インタプリタが前記複数のプロセッサユニット情報の全部を管理することにより前記複数のプロセッサユニットをシミュレーションすることを特徴とする請求項1記載の並列プロセッサ論理シミュレーション装置。

【請求項3】1つの命令インタプリタが1つのプロセッサユニット情報を管理し、前記プロセススケジューラが複数の命令インタプリタを制御することを特徴とする請求項1記載の並列プロセッサ論理シミュレーション装置。

【請求項4】前記回路シミュレータはイベント発生を前記プロセススケジューラに報告する手段を有し、前記イベントは、ネットワーク間データ転送送受信処理の完了、ネットワーク網の障害発生、ネットワーク網の所定サイクル連続した空状態の発生、ネットワーク網の所定サイクル連続したビジー状態の発生、タイマ割込みの検出であることを特徴とする請求項1記載の並列プロセッサ論理シミュレーション装置。

【請求項5】前記命令インタプリタはイベント発生を前記プロセススケジューラに報告する手段を有し、前記イベントは、ネットワークへの送信処理開始、ネットワークからの受信処理完了であることを特徴とする請求項1記載の並列プロセッサ論理シミュレーション装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、情報処理装置等を構成する論理回路の論理シミュレーション検証に係り、特に大規模な並列プロセッサシステムのネットワーク網の論理シミュレーションに関する。

【0002】

【従来の技術】従来、論理回路装置を構成するゲート単位の全基本論理素子を全て回路シミュレータに実装し、該回路シミュレータに命令インタプリタを接続し、プロ

2

グラム実行時、論理シミュレーション対象となるテスト命令列のみを回路シミュレータ側で実行し、該テスト命令列以外の命令列は命令インタプリタが代行実行する2階層論理シミュレーション方式が提案されていた。これに関する事例として、例えば、特開昭59-148971号公報が上げられる。

【0003】ネットワーク網で複数のプロセッサユニットが接続されている大規模な並列プロセッサシステムの論理シミュレーションを上記の従来技術により行なうと、プロセッサユニット論理回路部分を全て回路シミュレータに実装する為、回路シミュレータの論理実装制限により実装出来るプロセッサユニット台数が限定される。従って、全てのプロセッサユニットを実装した論理シミュレーションが出来ない。又、論理シミュレーション性能も実装規模が増える程遅くなり、シミュレーション時間が非常に長いと言う問題があった。

【0004】

【発明が解決しようとする課題】本発明の目的は、回路シミュレータにプロセッサユニットのプロセッサユニット制御論理部を実装せず、該プロセッサユニット制御論理部を全て命令インタプリタが代行実行し、ネットワーク論理及びプロセッサユニットのネットワーク制御論理部を回路シミュレータに実装することで上記の問題点を解決し、回路シミュレータの論理実装制限に関係無く、大規模な並列プロセッサシステムの論理シミュレーションを効率良く高速に実施することにある。

【0005】

【課題を解決するための手段】ゲートレベルの基本論理素子で実現される論理回路装置と等価な回路シミュレータを用いた並列プロセッサの論理シミュレーションにおいて、プロセッサネットワーク網論理及びプロセッサユニットのネットワーク制御論理部のみをシミュレーションする回路シミュレータ、データ転送レベルで複数のプロセッサユニット論理回路と等価な動作を擬似する命令インタプリタ、回路シミュレータ及び命令インタプリタを制御するプロセススケジューラ、回路シミュレータで発生したイベントを検出しプロセススケジューラへ制御を移す手段、命令インタプリタで発生したイベントを検出しプロセススケジューラへ制御を移す手段、ネットワーク構成情報を登録する手段とを有し、回路シミュレータにプロセッサユニット制御論理部を実装すること無しに、高速に大規模な並列プロセッサの論理シミュレーションを可能としたことにより達成される。

【0006】

【作用】本発明は並列プロセッサネットワーク論理のシミュレーションを、プロセッサユニット制御論理部を命令インタプリタがシミュレートする方式で実現している。必要最低限の論理実装で論理シミュレーションを実施できる為、低速な回路シミュレータの負担を低減し、シミュレーション全体の高速化を実現すると共に、並列

3

プロセッサシステム最大実装での論理シミュレーションを実現できる。

【0007】

【実施例】図1は本発明の論理シミュレーション対象となる並列プロセッサシステム構成の一例を示す図である。図1において110は並列プロセッサシステム、111はクロスバネットワーク網、112、113、114はプロセッサユニットである。

【0008】図中の矢印は、プロセッサユニット1及びiがクロスバネットワーク網を介して、データパケットをプロセッサユニット2に送信していることを示している。図中では、プロセッサユニット1からの送信データを、プロセッサユニット2が受信している状態で、プロセッサユニットiからの送信がクロスバネットワーク上で待たされる。この場合、プロセッサユニット1からの送信動作に対して、ビジーが発生し、処理待ちが起こる。又、プロセッサユニットiからの送信パケット長が短い場合、プロセッサユニットiからの送信動作は完了する（ビジーではなくなる）が、クロスバネットワーク網上にパケットデータが滞留している状態となる場合があり、この際、何らかの要因でプロセッサユニット1からプロセッサユニット2への送信でデッドロックが発生した場合、送受信関連イベントが発生しなくなる（これをネットワークが“空”の状態と呼ぶ）。

【0009】本発明は、このような種々の状態における並列プロセッサの動作を論理シミュレーションにより検証しようとするものである。

【0010】図2は本発明が適用される並列プロセッサ論理シミュレーションシステムの構成の一例を示す図である。

【0011】図2において200はホスト計算機システム、201はプロセススケジューラ、202は回路シミュレータ、203は回路シミュレータがシミュレートするネットワーク論理、204はメモリ、205は並列プロセッサのプロセッサユニットのシミュレートを行なう命令インタプリタ、206は命令インタプリタが制御する擬似メモリ、207は命令インタプリタが管理するプロセッサユニット情報、208は論理シミュレーション対象となる並列プロセッサのネットワーク構成情報を登録するファイルである。

【0012】ネットワーク構成情報ファイルには、クロスバネットワーク網構成を示す次元、各クロスバに接続されるプロセッサアドレス及び実装状態を示す有効/無効ビットが登録されている。命令インタプリタは本情報を参照し、シミュレートするプロセッサ台数及び実装アドレスを認識する。

【0013】図3は本発明を用いた論理シミュレーション形態の実施例を示す図である。

【0014】命令インタプリタ複数プロセッサモード(a)では、プロセススケジューラ(301)下に、回

4

路シミュレータ(302)と命令インタプリタ(303)を一台づつ配して、命令インタプリタが複数のプロセッサユニット情報(304)を有し、複数のプロセッサユニットをシミュレートする。命令インタプリタ単一プロセッサモード(b)では、プロセススケジューラ下に、回路シミュレータ一台とシステム構成上必要となるプロセッサユニット台数分の命令インタプリタ1(305)から命令インタプリタN(306)迄を配して、各命令インタプリタがネットワーク網に接続される各プロセッサユニットをシミュレートする。両モード共プロセススケジューラは、ホスト計算機のタイマを用いて、各回路シミュレータ及び、命令インタプリタに順次サービスを行なうが、命令インタプリタ側でネットワークへの送信処理開始時点、ネットワークからの受信処理完了時点で命令インタプリタから回路シミュレータに一旦制御を移す。又、回路シミュレータ側で、ネットワーク間データ転送送受信処理が完了した時点、ネットワークで障害発生を検出した時点、ネットワーク網が所定サイクル連続して空状態となった時点、ネットワーク網が所定サイクル連続してビジー状態となった時点、タイマ割込みを検出した時点では、同様に回路シミュレータから対象プロセッサユニットのシミュレートを担当している命令インタプリタ側に制御を移す。

【0015】図4は本発明の一実施例のプロセッサユニット情報を示す図である。400は対象プロセッサユニットのレジスタ情報、401は対象プロセッサユニットの実行環境情報であり、命令実行アドレスを示すプログラムカウンタ(402)、制御移行事象を登録するイベントフラグ(403)等が格納されている。404はメモリ管理テーブルのアドレスを格納する。

【0016】メモリ管理テーブルは現在アロケートされたページ、すなわち現在までにデータの書き込みが発生したページの論理アドレス及び、実際に命令インタプリタがホスト計算機上に確保した物理メモリアドレスの対応をページ単位に管理する。

【0017】図5は本発明の命令インタプリタのメモリ管理の一実施例を示す図である。

【0018】命令インタプリタは、命令実行中にメモリへの書き込みが発生した場合(500)、対象となるメモリページと書き込みページアドレス(501)のみホストマシンに動的に確保する。メモリアドレス管理テーブルに、確保したページの論理アドレス、ホスト計算機上に確保した該当ページ領域の物理アドレス、及びダーティフラグが登録される(502)。ダーティフラグとは、ネットワーク論理へ状態を遷移する際、前回の状態遷移以降書き換えが該当ページに対して発生したか否かを示す1バイトのフラグバイトであり、ダーティフラグの立っているページのみが、状態遷移の際のデータコピー対象となる。ダーティフラグは、状態遷移直前に全てクリア(503)される。回路シミュレータ側メモリにも

同様のダーティラグがあり、命令インタプリタへの状態遷移時に同様の処理が行なわれる。この処理により、必要最小限のホスト計算機のメモリのみを使用した高速なデータ転送が可能となる。図2の構成で回路シミュレータ側がメモリ(204)を持たない構成の場合、状態遷移時にデータコピーが発生しないため、ダーティフラグのクリア(503)は行なわない。

【0019】図6は本発明を用いた命令インタプリタ複数プロセッサモードでの並列プロセッサの論理シミュレーションの一実施例の流れを示す概略フローチャートである。プロセススケジューラはシミュレーションが開始される(600)とネットワーク構成情報ファイルよりネットワーク構成情報を取得し、所定領域に該情報を設定(601)した後、回路シミュレータを起動する(602)。回路シミュレータは、該ネットワーク構成情報に従いパワーオンリセット処理(603)を実行しネットワークを初期化した後、プロセススケジューラを介して命令インタプリタを起動する(604)。命令インタプリタは、該ネットワーク構成情報に従いネットワーク接続台数分のプロセッサメモリに対象プログラムをロード(605)する。

【0020】命令インタプリタは、プロセッサユニット環境情報を1命令実行ごとに切り替え、ネットワーク接続台数分のプロセッサユニットを擬似実行する(606)。全プロセッサユニット環境情報を1命令実行ごとに切り替え、ネットワーク接続台数分のプロセッサユニットの1命令の処理が完了した時点でネットワーク制御移行イベント発生有無のチェックを行なう(607)。ネットワーク制御移行イベントが発生しない場合は、全プロセッサユニットで実行されているプログラムが全て終了する迄前記処理を繰り返す(612)。全プロセッサユニットで実行されているプログラムが全て終了した場合、制御をプロセススケジューラに戻し、プロセススケジューラはシミュレーションを終了させる(613)。ネットワーク制御移行イベントが発生した場合は、プロセススケジューラを介して制御を回路シミュレータへ移す。この際、プロセススケジューラは命令インタプリタが管理しているプロセッサユニットのメモリデータを、論理フォーマットに変換し、回路シミュレータ側メモリへコピーする(608)。回路シミュレータは、制御移行時点からのネットワーク網のシミュレーションを実行する(609)。ネットワーク論理でプロセッサ制御移行イベントが発生した場合(610)、プロセススケジューラを介して、命令インタプリタ側へ制御を移す。この際、プロセススケジューラは回路シミュレータ側メモリデータをプログラマブルデータフォーマットに変換し、命令インタプリタが管理しているプロセッサユニットのメモリへコピーする(611)。図2の構成で回路シミュレータ側がメモリ(204)を持たない構成の場合、回路シミュレータ側でメモリアクセスが発

生した際に、プロセススケジューラを介してインタラティブに命令インタプリタ側メモリをアクセスする。このため、回路シミュレータ及び命令インタプリタ間の状態遷移(604及び611)の際、データ変換/コピーは実施しない。

【0021】図6では、命令インタプリタ複数プロセッサモードの場合であったが、命令インタプリタ単一プロセッサモードの場合では、システム構成上必要となるプロセッサユニット台数分の命令インタプリタがプロセススケジューラに起動される。以降の処理手順は、プロセススケジューラが回路シミュレータ及び複数の命令インタプリタに対して、均等にタイムスライス制御でサービスを行なう以外は、命令インタプリタ複数プロセッサモードと同様である。但し、各命令インタプリタは、単一プロセッサユニットを擬似実行している為、1命令実行毎のプロセッサユニット環境情報の切り替えは必要ない。

【0022】又、本実施例では、ネットワーク制御イベントが発生した際に、回路シミュレータ及び命令インタプリタ間の状態遷移を実施しているが、回路シミュレータ側では1サイクルの状態変化、命令インタプリタ側では全てのプロセッサ環境で1命令の処理が終了した時点で、無条件に制御移行する方法もある。

【0023】図7は、状態遷移時又は、回路シミュレータ側からのメモリアクセス時に対象となるメモリのプログラマブルデータフォーマット(700)、論理フォーマット(701)及び変換時のデータ位置対応の一例を示す。プログラムが直接扱うデータ形式がプログラマブルデータフォーマット、実際の論理上は、メモリRAMの物理配置(インタリーブ等)やアクセス形式、ハミングコード付加(702)によりデータ形式が異なる。これを論理フォーマットと呼ぶ。

【0024】

【発明の効果】以上説明した様に本発明によれば、回路シミュレータの論理実装制限によらない大規模な並列プロセッサの論理シミュレーションを可能とし、又従来に比して高速な論理シミュレーションを容易に実現出来る。

【図面の簡単な説明】

【図1】本発明の論理シミュレーション対象となる並列プロセッサシステム構成の一例を示す図である。

【図2】本発明が適用される並列プロセッサ論理シミュレーションシステムの構成の一例を示す図である。

【図3】本発明を用いた論理シミュレーション形態の一例を示す図である。

【図4】本発明のプロセッサユニット情報の一例を示す図である。

【図5】本発明の命令インタプリタのメモリ管理の一例を示す図である。

【図6】本発明を用いた論理シミュレーションの一実施

7

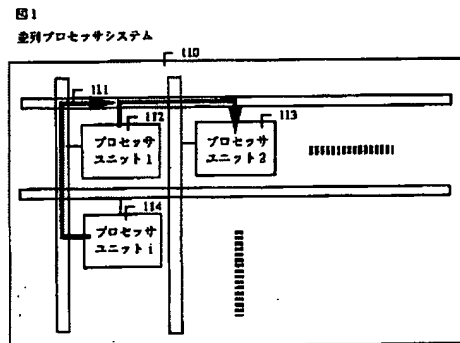
例の動作の流れを示す概略フローチャートである。

【図7】本発明が適用されるメモリのデータフォーマット及び変換時のデータ位置の対応の一例を示す図である。

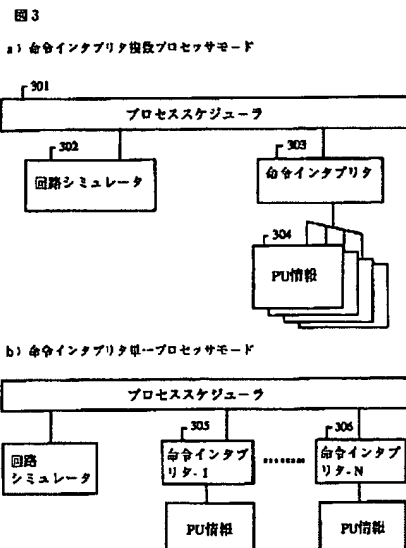
【符号の説明】

200：ホスト計算機システム 201：プロセス
スケジューラ

【図1】



【図3】



8

202：回路シミュレータ

203：回路シミュレータがシミュレートするネットワーク論理

204：メモリ

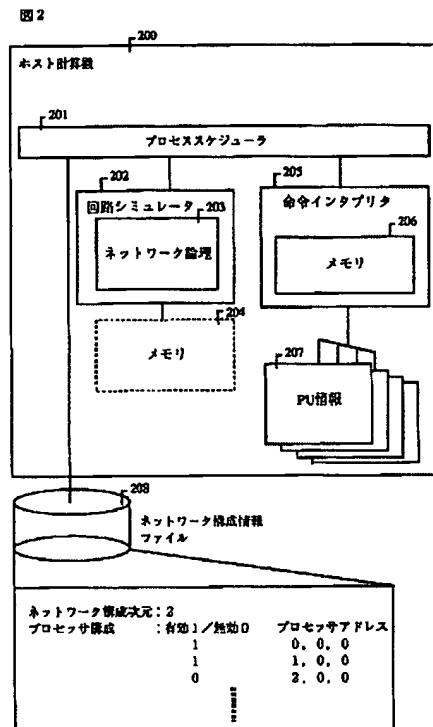
205：命令インタプリタ

206：命令インタプリタが制御する擬似メモリ

207：プロセッサユニット情報

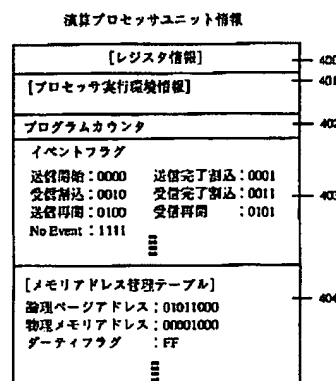
208：ネットワーク構成情報を登録するファイル

【図2】



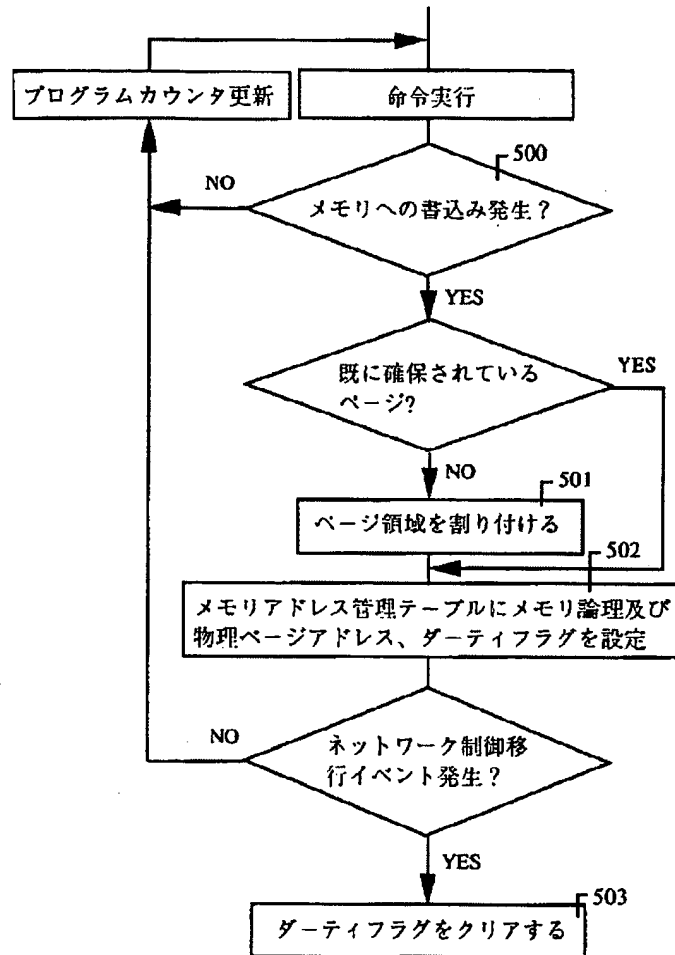
【図4】

図4



【図5】

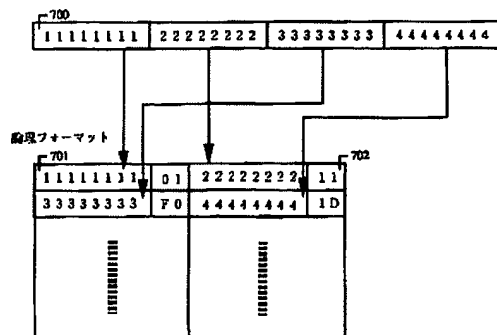
図5



【図7】

図7

プログラマブルデータフォーマット



【図6】

図6

